

PROGETTAZIONE

PROGETTAZIONE ARCHITETTURALE

REQUISITI NON FUNZIONALI

Nella tabella dei vincoli sono emersi tre requisiti fondamentali:

- Usabilità
- Sicurezza
- Disponibilità

Considerando il sistema che deve essere sviluppato, i requisiti di maggiore importanza sono l'usabilità e la disponibilità.

La disponibilità è fondamentale in quanto malfunzionamenti o perdite di dati durante l'interazione coi clienti porterebbero a gravi conseguenze oltre che economiche anche di immagine. Il vincolo dell'usabilità è importante poiché non sappiamo quanta esperienza abbiano i clienti con i sistemi informatici. I due requisiti non sono in conflitto tra di loro, infatti l'usabilità riguarda principalmente le interfacce presentate ai clienti e ai dipendenti del negozio, mentre la disponibilità impone un occhio di riguardo sulla replicazione dei componenti critici. Da non sottovalutare è l'aspetto della sicurezza, in particolare la protezione di dati sensibili dei clienti e i dati riservati dell'azienda.

SCelta DELL'ARCHITETTURA

Dal punto di vista architetturale, l'architettura più idonea per questo tipo di sistema è un'[architettura client/server a tre livelli](#).

L1 – Client:

Si è scelto di sviluppare un client che in base all'utente che lo userà effettuando il login presenterà differenti funzionalità:

- Per il Cliente
- Per il Cassiere
- Per il GestoreNegozio
- Per il GestoreSicurezza
- Per l'AmministrazioneInventario

L2 – Server

Per il lato Server si hanno diversi server:

- Uno per le funzioni del Cassiere
- Uno per le funzionalità relative ai clienti
- Uno per la gestione dei Log
- Uno per l'amministratore dell'inventario

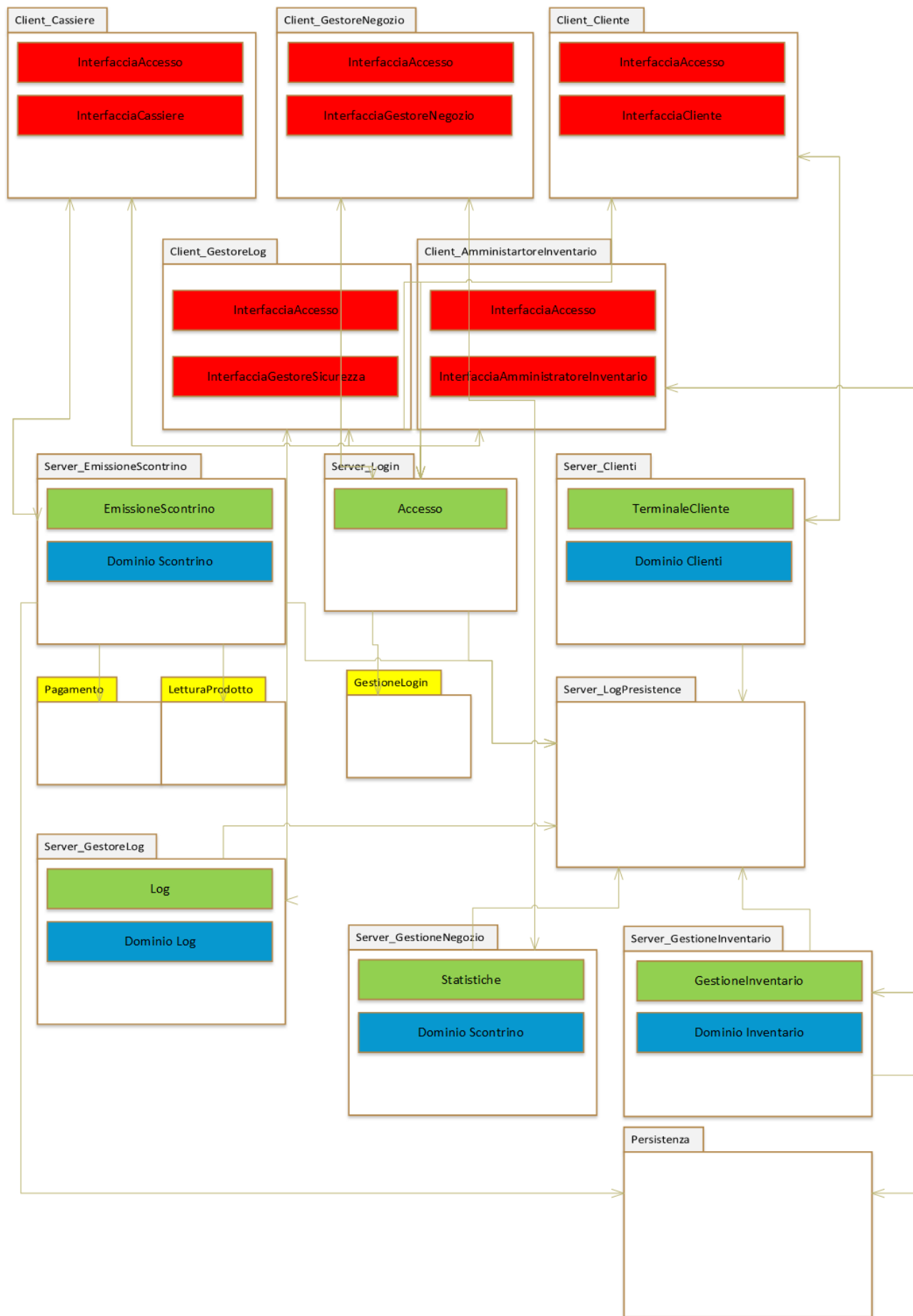
- Uno per il login

Separare i Server è importante in quanto nel caso di guasto solo una funzionalità smetterà di funzionare. Inoltre il server sarà facilmente sostituibile da macchine duplicate periodicamente.

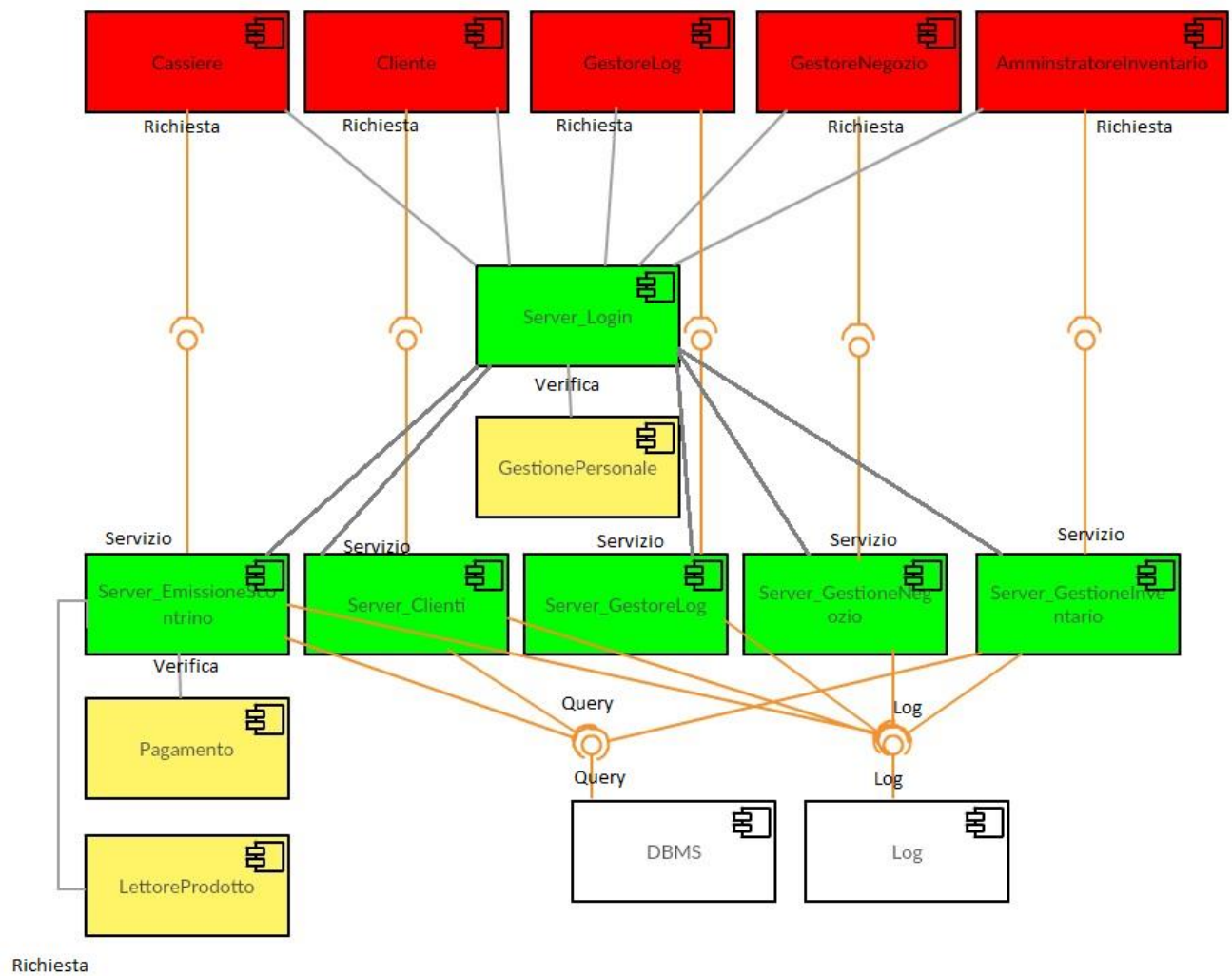
L3 – Persistenza

Per la gestione della persistenza avremo un server dedicato con un opportuno database con tutte le informazioni relative all'inventario. Sarà possibile ripristinare le macchine o i sistemi danneggiati ad uno stato precedente integro.

Nella figura sottostante è riportata l'Architettura del Sistema organizzata attraverso un diagramma dei package:



Nella figura sottostante è riportata l'architettura del sistema organizzata attraverso un diagramma dei componenti:

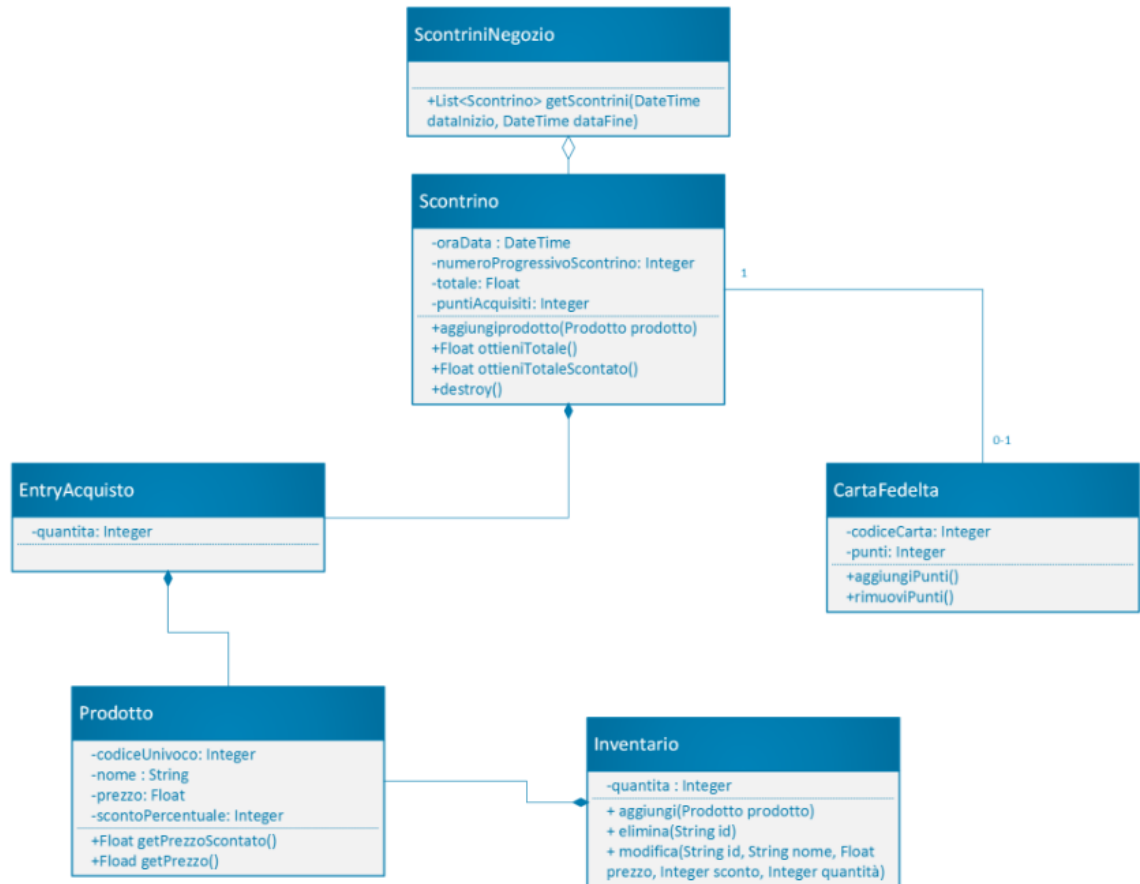


Progettazione di Dettaglio

Nel seguito si riportano i diagrammi di dettaglio delle parti del Sistema.

Struttura

Diagramma di Dettaglio: EmissioneScontrino



Abbiamo spostato L'inventario in questo diagramma in quanto è una composizione di prodotti. La funzione ottieneTotale è stata separata in ottieniTotale() ed ottieniTotaleScontato()

Diagramma di Dettaglio: Log

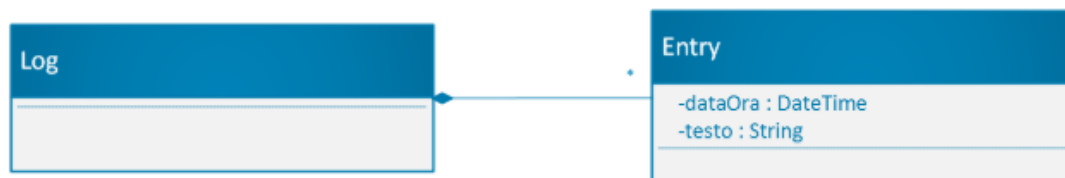
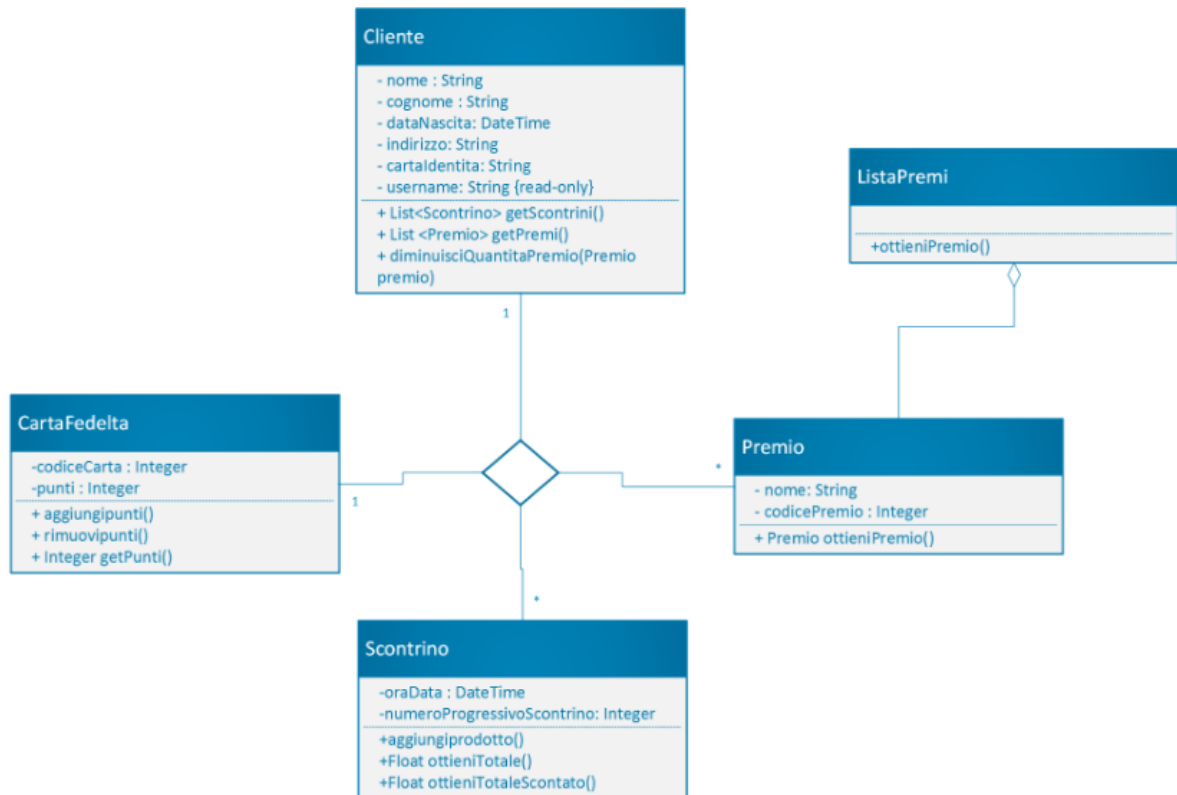


Diagramma di Dettaglio: operazioniClienti



Username è read-only in quanto dopo che è stato definito in registrazione non è più modificabile.

Diagramma di Dettaglio: interfacce nei Server

Grazie all'uso di interfacce possiamo dipendere su astrazioni e non su oggetti concreti applicando così il **Dependency Inversion Principle**. I moduli di alto livello infatti non dipendono da quelli di basso livello ma entrambi dipendono da astrazioni.

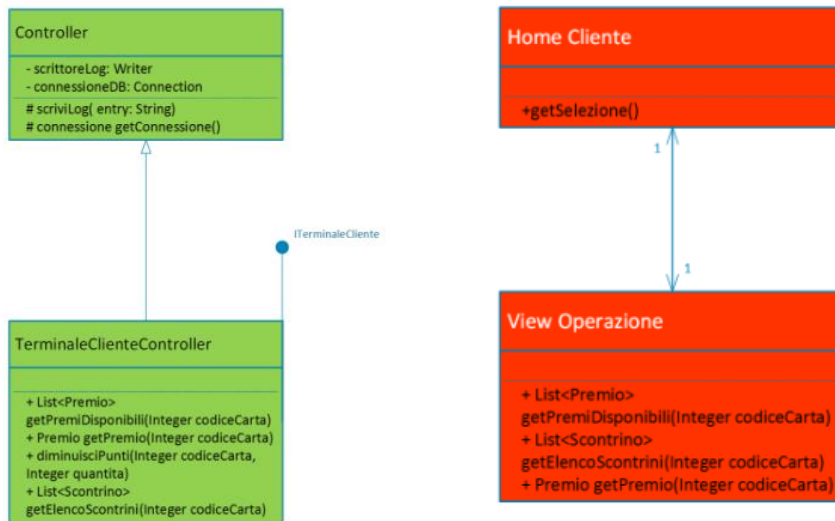


Per quanto riguarda la scrittura su Log e le connessioni al Database si è creato CONTROLLER in maniera analoga a quella dell'esempio proposto a lezione non essendo il log centrale nel nostro progetto. Quando sarà presente un nuovo Log , Controller con la sendLog del client() lo manderà al server che si occuperà di salvarlo su file.

In ogni diagramma di dettaglio alle varie home sarà collegata la classe "ClientLog". Per non duplicarla in ogni diagramma la riporteremo qui sotto:



Diagramma di Dettaglio: Terminale Client



Di seguito si riportano gli schemi delle interfacce del cliente.

OTTIENI PREMIO

ELENCO SCONTRINI

SCONTRINO	DATA	IMPORTO	PUNTI ACQ.
184369	28/10/2017	67,34 €	67
193748	15/11/2017	56,87 €	57
199283	28/11/2017	45,90 €	46
201349	09/12/2017	82,31 €	82
210293	23/12/2017	41,75 €	42
215463	30/12/2017	39,14 €	39

OTTIENI PREMIO

ELENCO SCONTRINI

PREMIO	PUNTI
<input type="radio"/> Pentole	900
<input type="radio"/> Servizio di piatti	600
<input type="radio"/> Set di bicchieri	500
<input type="radio"/> Orologio da parete	500
<input type="radio"/> Set di posate	400
<input checked="" type="radio"/> Materasso	1000

SALDO

2400

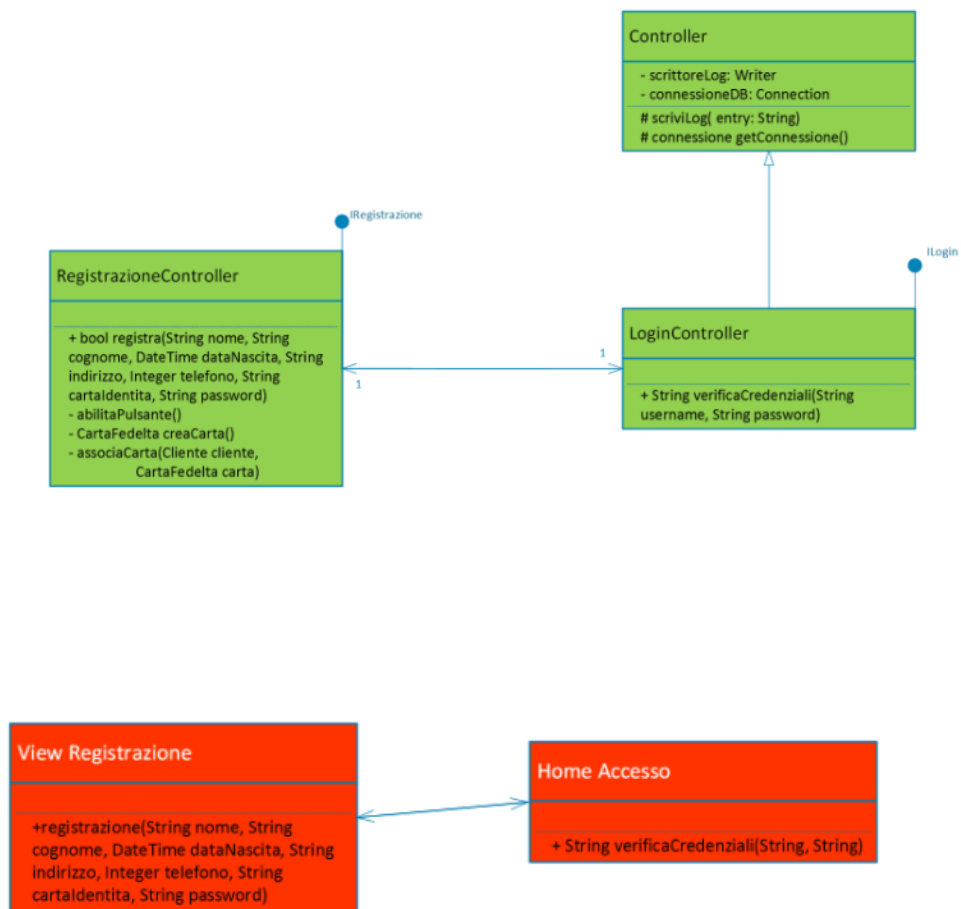
-1000

SALDO RIMANENTE

1400

OTTIENI PREMIO

Diagramma di Dettaglio: Login



Abilitapulsante abilita il pulsante solo se rileva di essere nel Terminale del Client.

Di seguito si riportano gli schemi delle interfacce del login.

✕

LOGIN

Username

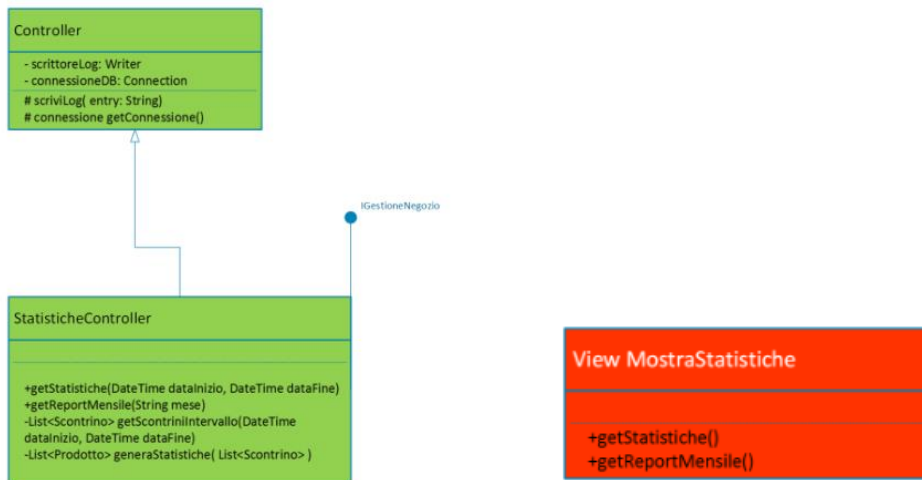
Password

✕

Nome	<input type="text"/>
Cognome	<input type="text"/>
Data di nascita	<input type="text"/> / <input type="text"/> / <input type="text"/>
Città/Provincia	<input type="text"/> <input type="text"/>
CAP	<input type="text"/>
Via	<input type="text"/>
Telefono	<input type="text"/>
N° carta identità	<input type="text"/>
Password	<input type="password"/>

REGISTRA

Diagramma di Dettaglio: Statistiche

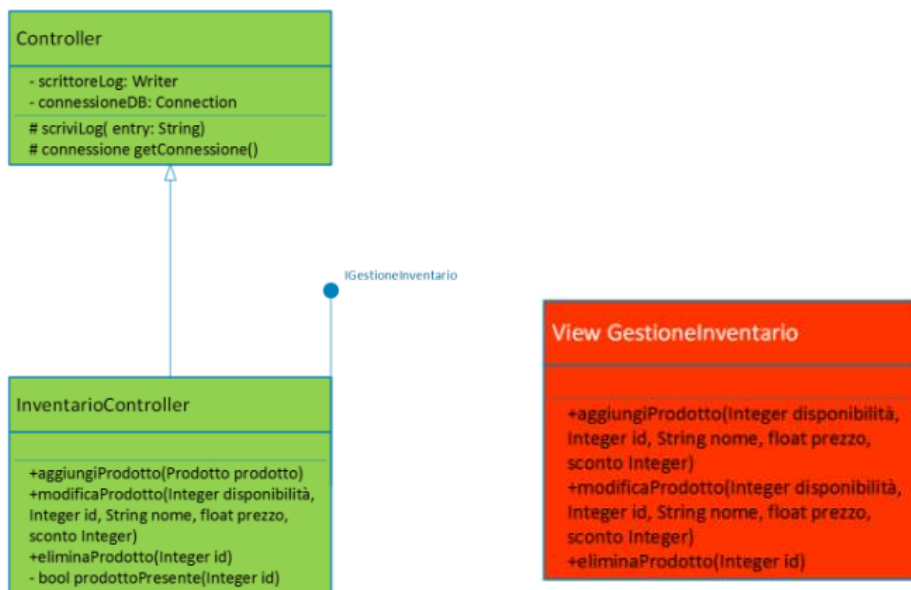


Di seguito si riportano gli schemi delle interfacce del gestore Negozio.

Report mensile	Per data	✕
----------------	----------	---

Report mensile	Per data	Report mensile	Per data
Data di inizio	<input type="text"/>	MESE	<input type="text"/>
Data di fine	<input type="text"/>		
<div></div>		<div></div>	

Diagramma di Dettaglio: Inventario



Di seguito si riportano gli schemi delle interfacce dell'inventario.

	MODIFICA	AGGIUNGI	RIMUOVI			
	QUANTITÀ	ID	NOME	PREZZO	SCONTO	
<input type="radio"/>	40	41236	Cioccolata Milka	1,72 €	0 %	
<input type="radio"/>	50	71324	Patatine Lay's	1,54 €	0 %	
<input type="radio"/>	85	21452	Latte Granarolo	1,24 €	10 %	
<input type="radio"/>	28	10097	Pollo Amadori	3,39 €	20 %	
<input type="radio"/>	61	52947	Birra Moretti	1,02 €	0 %	
<input checked="" type="radio"/>	47	28390	Fazzoletti Tempo	1,90 €	15 %	

QUANTITÀ
 ID
 NOME
 PREZZO
 SCONTO

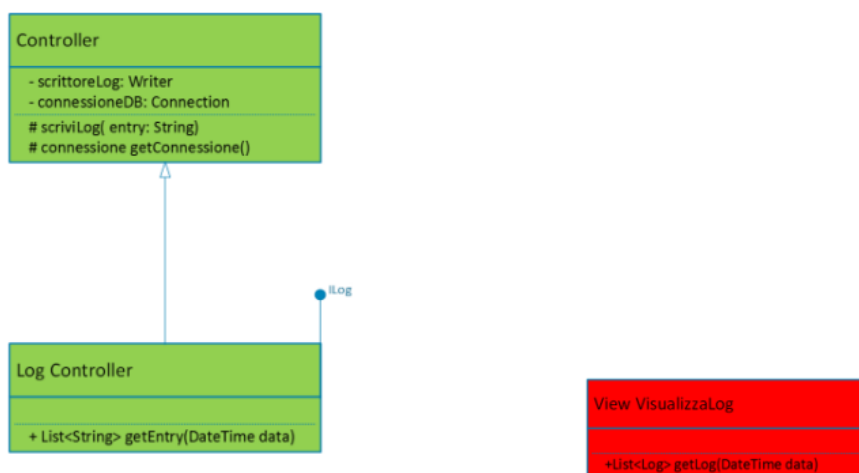
AGGIUNGI

QUANTITÀ
 ID
 NOME
 PREZZO
 SCONTO

MODIFICA

Per rimuovere basta selezionare l'oggetto e cliccare sul tasto RIMUOVI OGGETTO.

Diagramma di Dettaglio: Gestore Log



È risultato necessario aggiungere un filtro per la ricerca dei Log: è stato deciso di presentare solo i Log per la data inserita dal Gestore della Sicurezza.

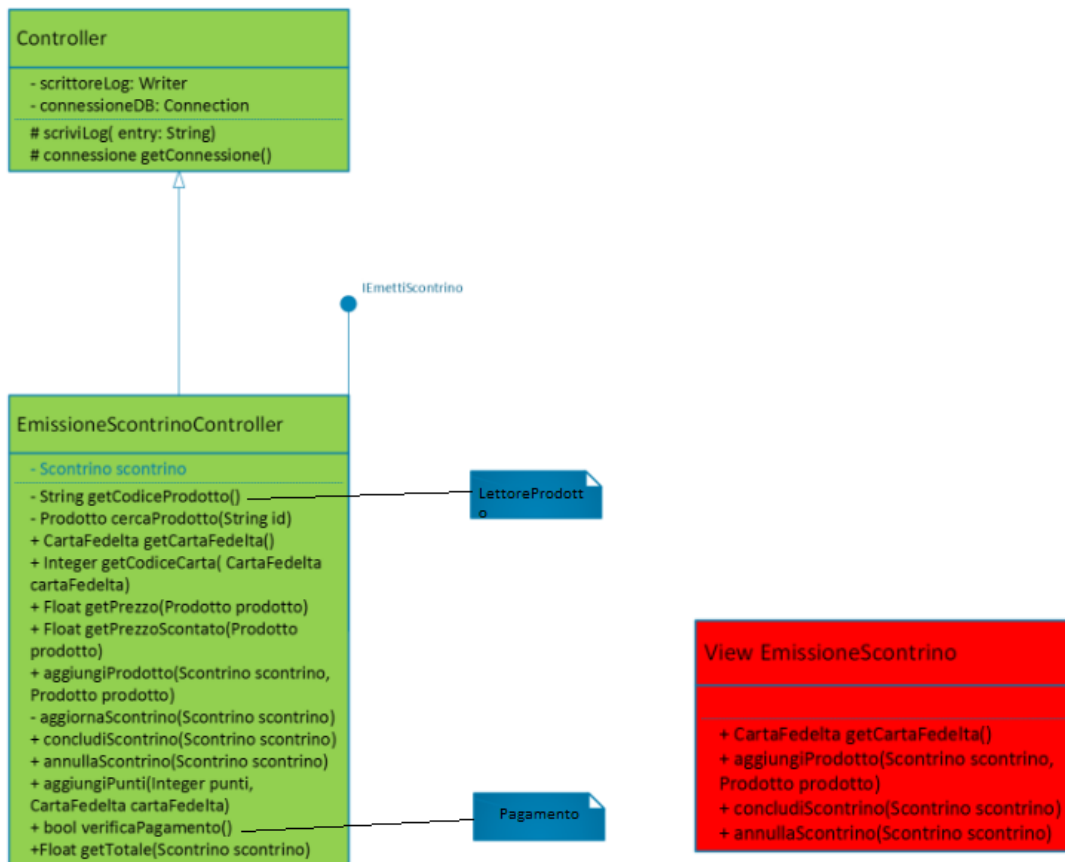


DATA

21/12/2017

ORA	TESTO	MACCHINA

Diagramma di Dettaglio: Emetti Scontrino



Le funzioni getCodiceProdotto() e verificaPagamento() rispettivamente interagiranno con le interfacce ILettoreProdotto e IPagamento dei sistemi esterni.

Carta ☒ Sì ☐ No ☒

9405148539

Nome

n°	NOME	PREZZO
1	Latte Granarolo	1,24 €
3	Birra Moretti	1,02 €
2	Patatine Lay's	1,54 €
1	Pollo Amadori	3,39 €

TOTALE

Interazione

Diagramma di sequenza: BilancioMensile

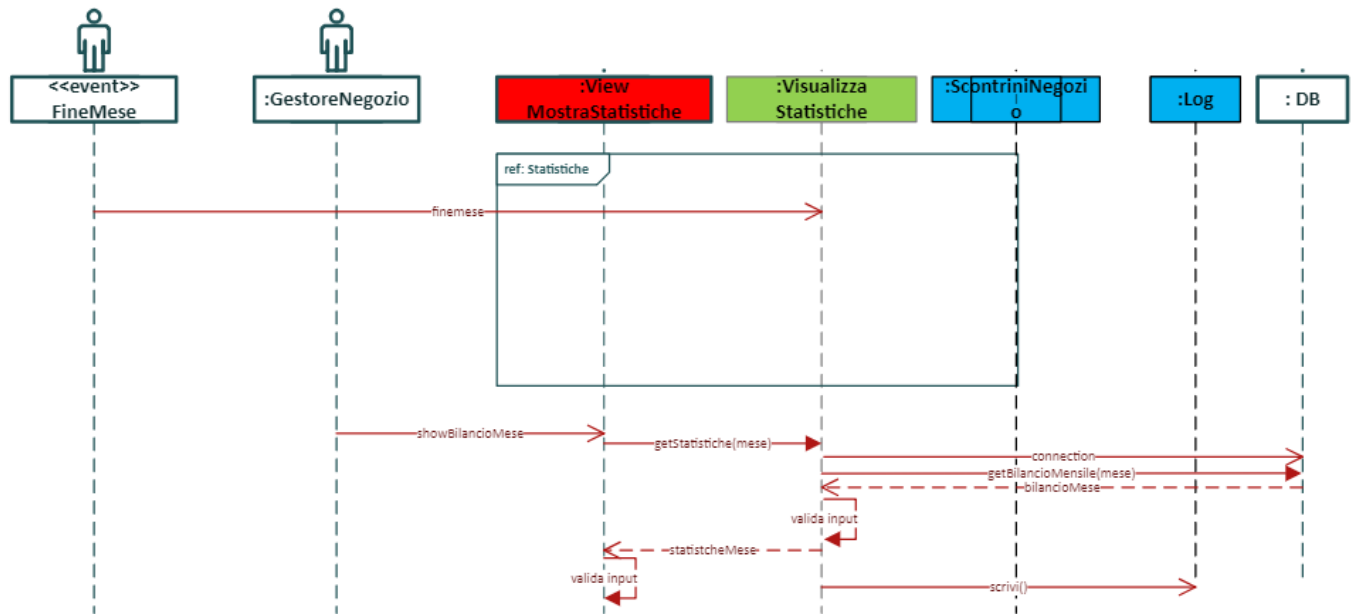
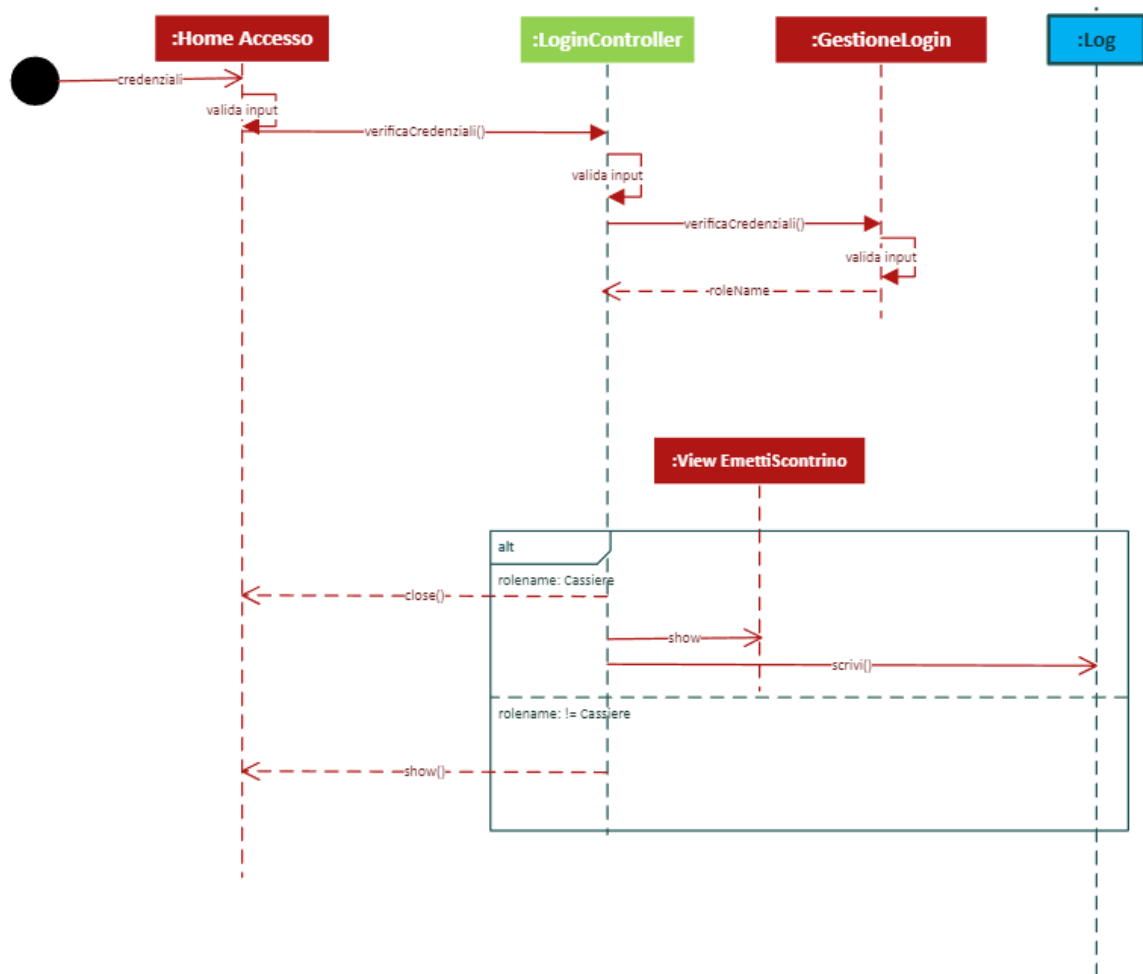


Diagramma di sequenza: Login Cassiere



Viene rappresentato il caso di un login andato a buon fine in quanto la sequenza temporale è uguale per tutte le persone che interagiscono con il sistema a differenza della view presentata dopo.

Diagramma di sequenza: elencoScontrini

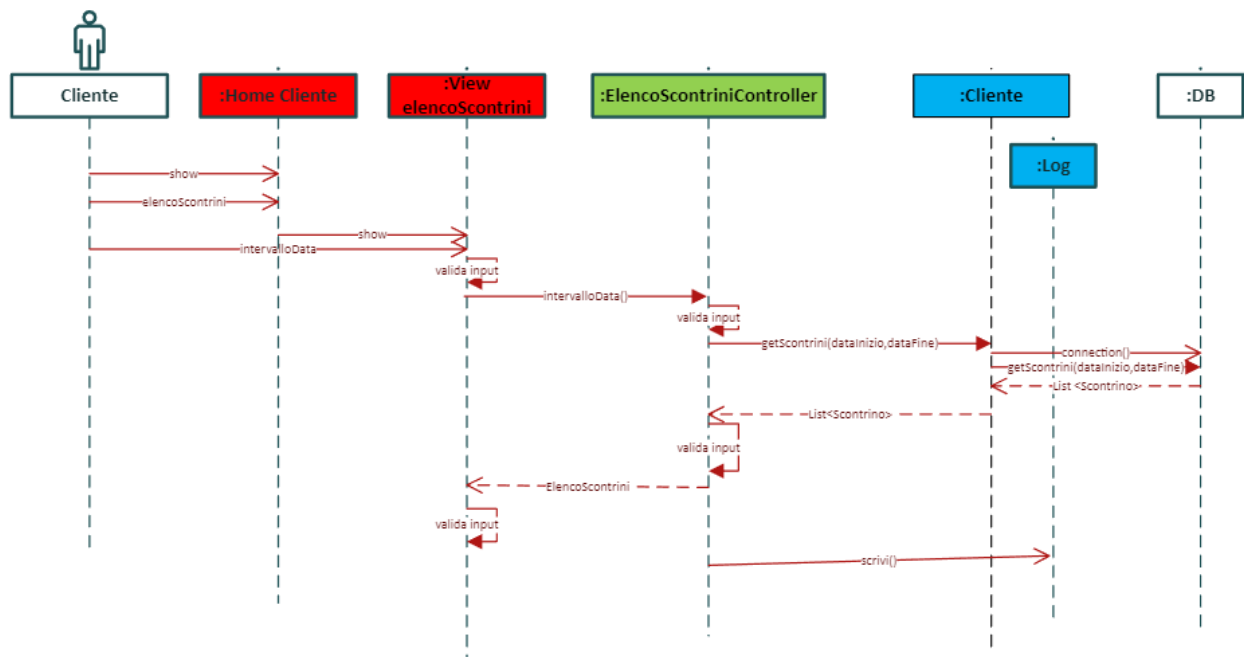


Diagramma di sequenza: emissioneScontrino

Le frecce di validazione input sono sottointese per questioni di spazio nel diagramma.

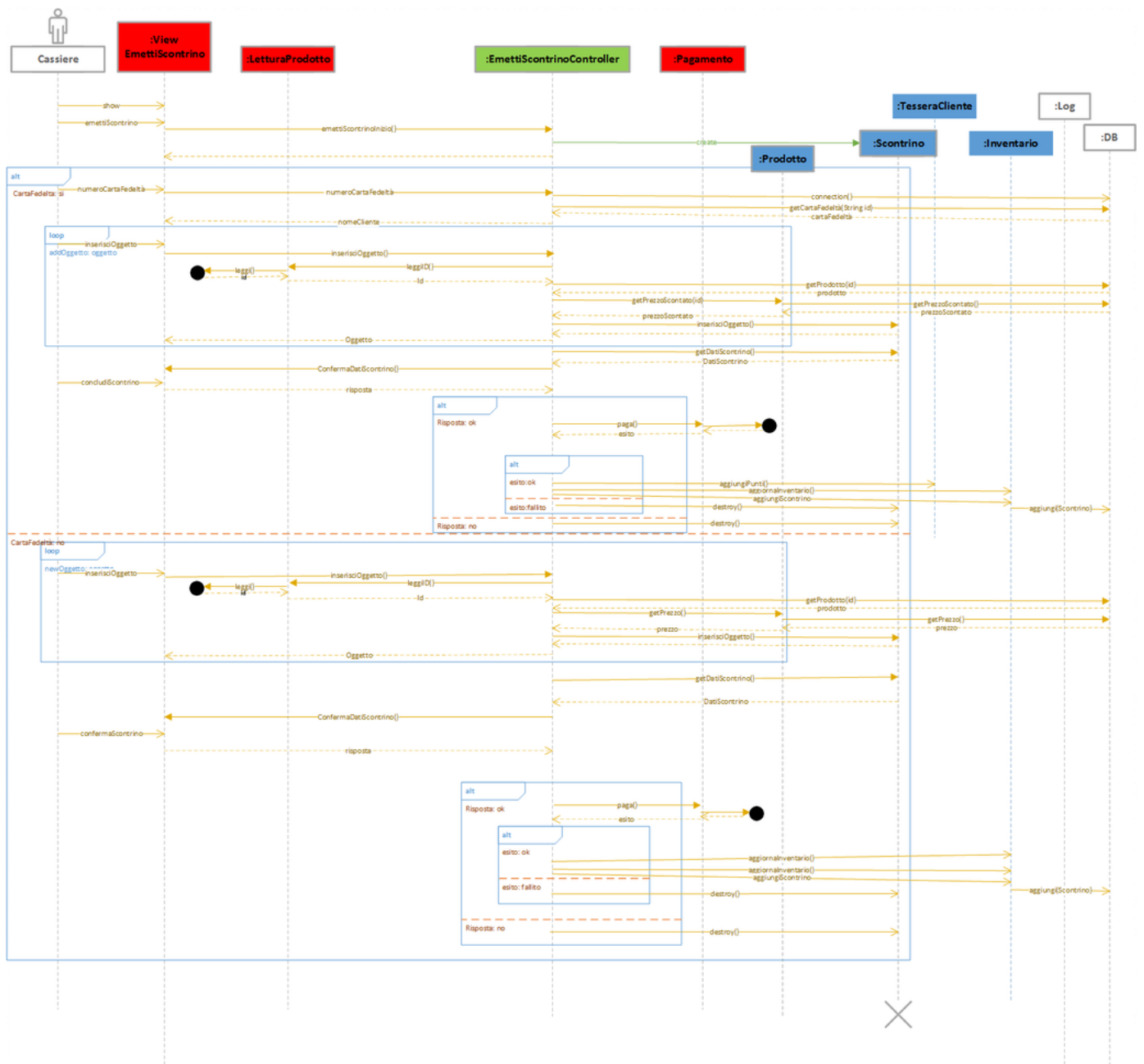


Diagramma di sequenza: gestioneInventario

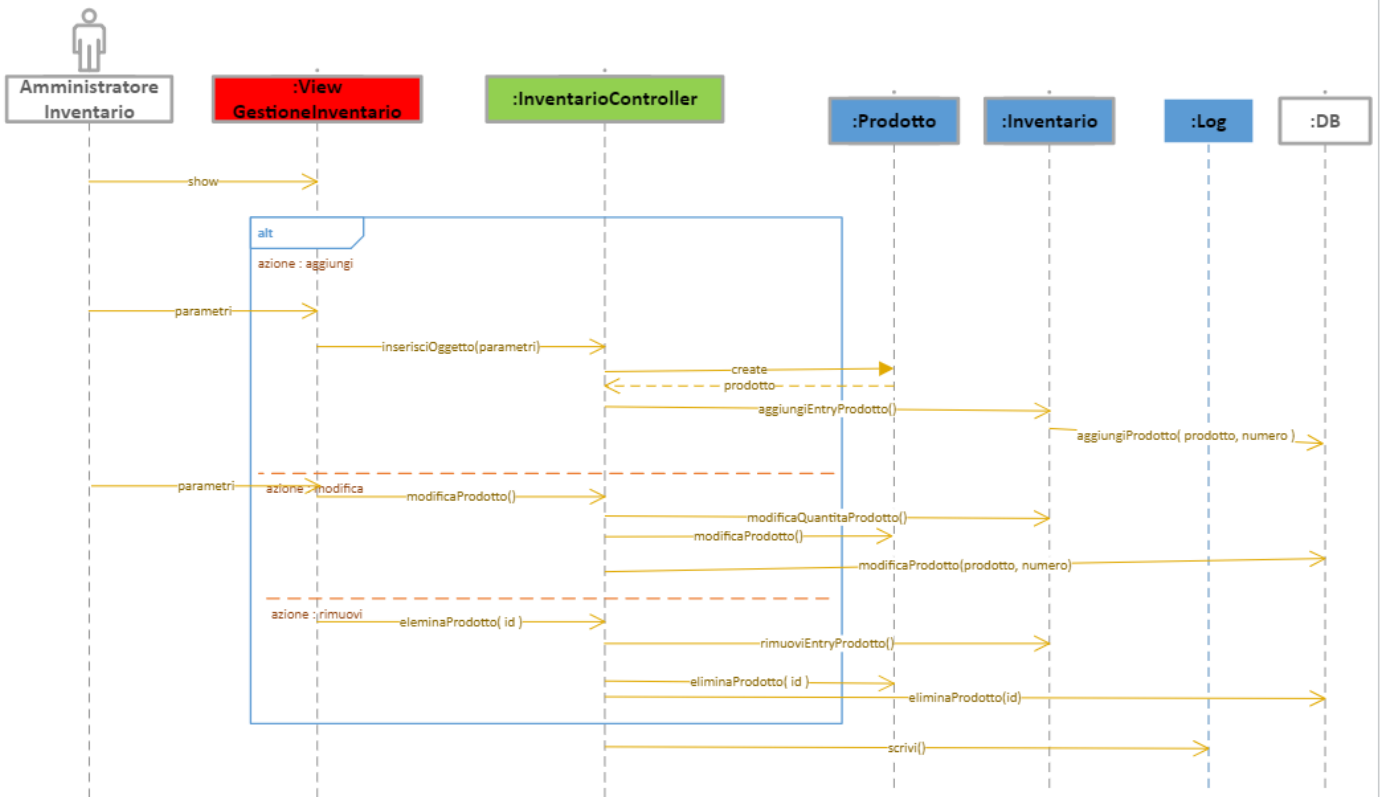


Diagramma di sequenza: gestioneInventario

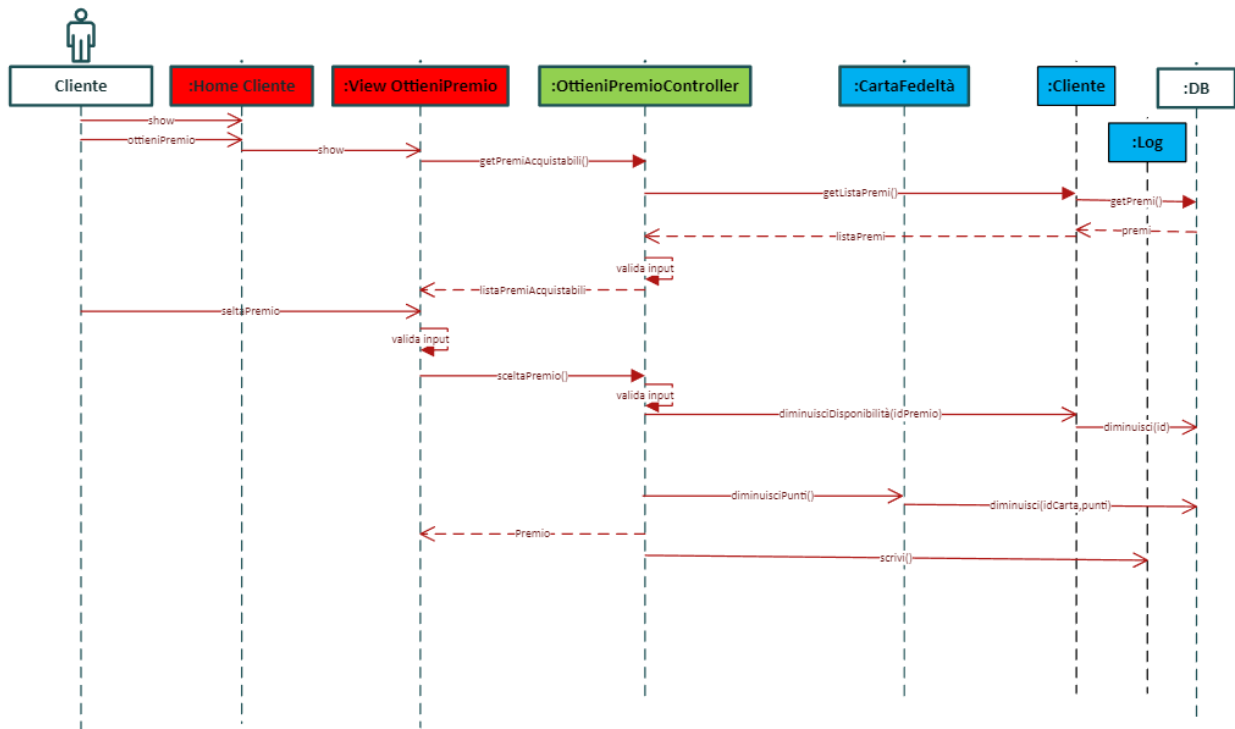
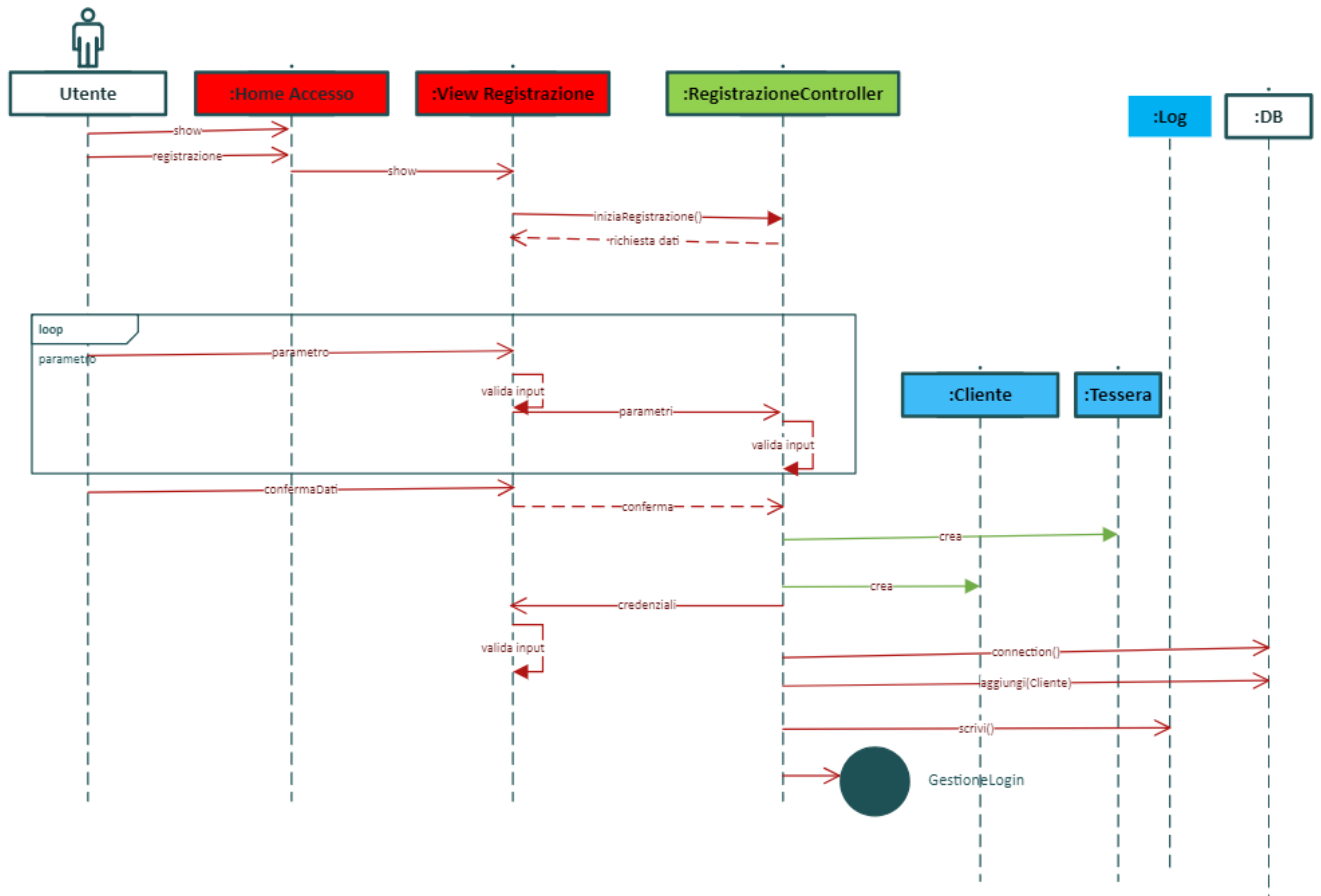
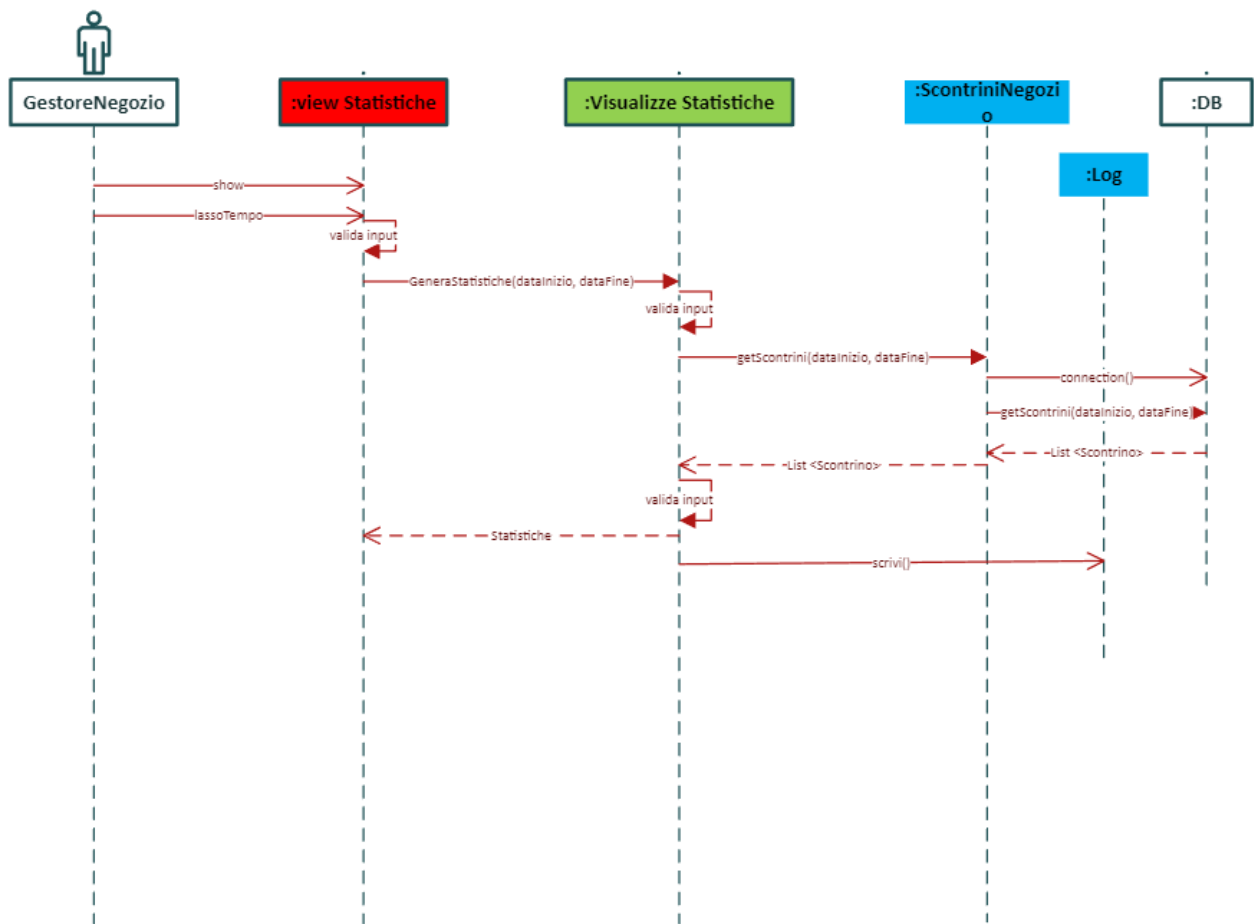


Diagramma di sequenza: Registrazione



Vengono passati al sistema esterno di GestioneLogin l'username e la password per consentire i login futuri.

Diagramma di sequenza: Statistiche



Comportamento

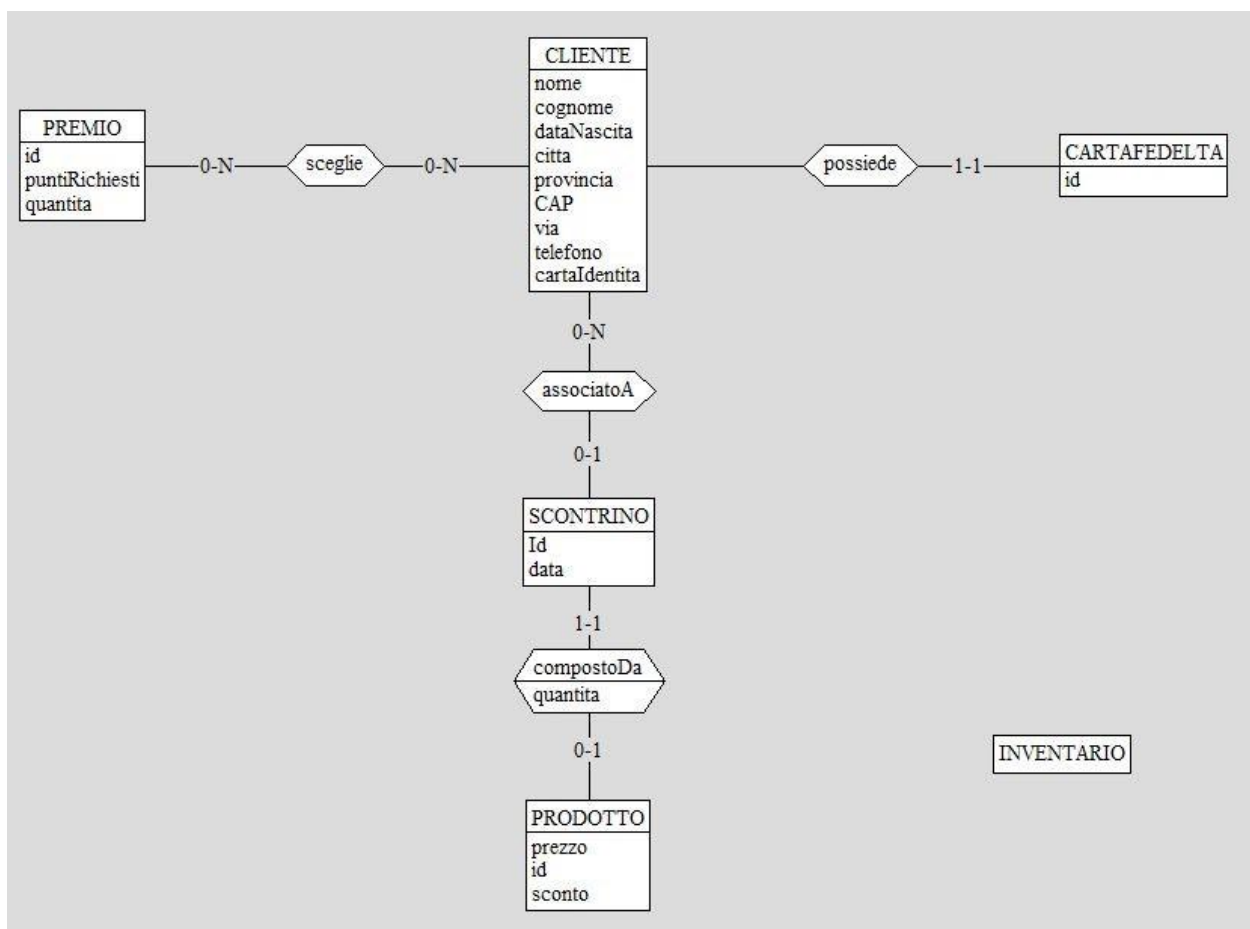
Non sono risultati necessari diagrammi di stato, invece è stato fatto un diagramma delle attività nell'analisi del problema che risulta sufficiente.

Progettazione della persistenza

Si è deciso di adottare un Database per quanto riguarda la persistenza dell'inventario degli oggetti del negozio, degli scontrini, dei report mensili e dei premi ottenibili tramite punti. Questa scelta è necessaria in quanto vengono trattati un considerevole numero di dati eterogenei che cambiano spesso e devono essere aggiornati. Tramite questa scelta sarà possibile anche ripristinare i dati ad una versione precedente in caso di malfunzionamento.

Il database verrà messo in sicurezza fisicamente e verranno effettuati backup giornalieri per prevenire eventuali malfunzionamenti.

Diagramma E-R:



Per il log invece ci sembra più giusto usare una scrittura su file (propriamente protetto) in quanto per ogni log si scrive una riga. L'accesso in questo modo sarà veloce ed ogni riga sarà in ordine cronologico.

Il file sarà leggibile solo dal terminale del gestore dei Log.

Formato File Log:

DataOra roleName username operazione

Progettazione del collaudo

```
[Test Fixture]
public class TestSCliente
{
    private Cliente _cliente

    [SetUp]
    public void ClienteSetUp(){
        _cliente = new Cliente("Francesco", "Rossi", new DateTime(10/06/1976), "Citta:Bologna -
        Provincia:BO - CAP:40123 - Via:via Saragozza 21", "ER2043293", "franc.ros");
    }

    [Test]
    public void TestMethod(){
        Assert.That(_cliente.getNome(), Is.EqualTo("Francesco"));
        Assert.That(_cliente.getCognome(), Is.EqualTo("Rossi"));
        Assert.That(_cliente.getDataNascita(), Is.EqualTo(new DateTime(10/06/1976)));
        Assert.That(_cliente.getIndirizzo(), Is.EqualTo("Citta:Bologna - Provincia:BO -
        CAP:40123 - Via:via Saragozza 21"));
        Assert.That(_cliente.getCartaIdentita(), Is.EqualTo("ER2043293"));
        Assert.That(_cliente.getUsername(), Is.EqualTo("franc.ros"));
    }
}
```

```

[Test Fixture]
public class TestScontrino
{
    private Scontrino _scontrino

    [SetUp]
    public void ScontrinoSetUp(){
        _scontrino = new Scontrino(DateTime.now(), 0000012321);
    }

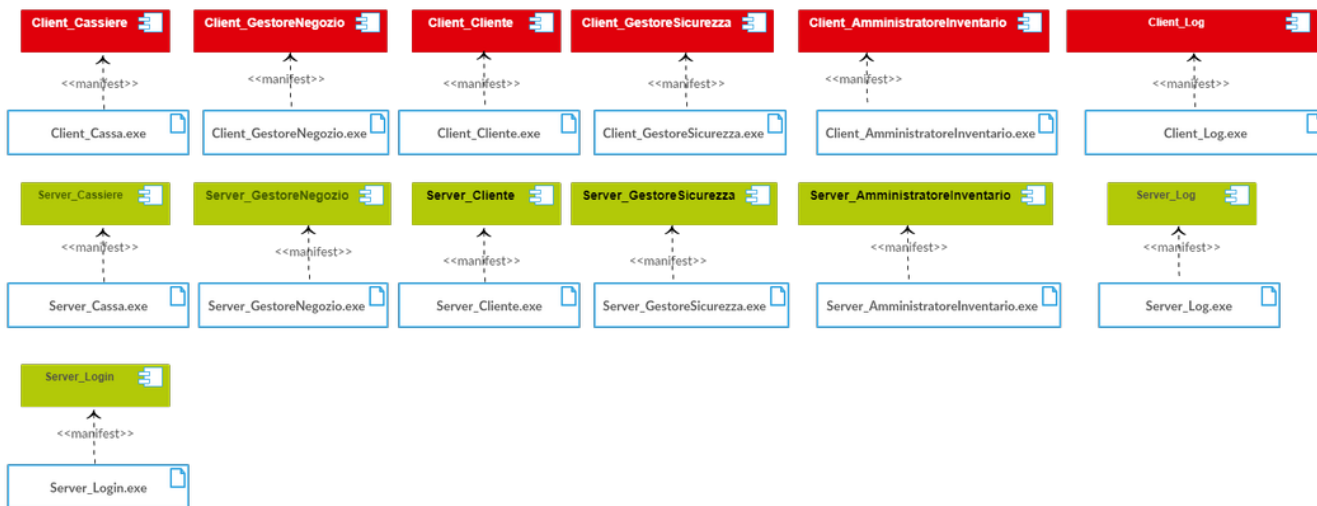
    [Test]
    public void TestMethod(){
        Assert.That(_scontrino.getNumeroProgressivo(), Is.EqualTo(0000012321));
    }

    [Test]
    public void TestMethod(){
        _scontrino.aggiungiProdotto(New Prodotto("AB123", "Bagnoschioma", 3.20, 0));
        _scontrino.aggiungiProdotto(New Prodotto("CD294", "Fazzoletti", 1.00, 0));
        _scontrino.aggiungiProdotto(New Prodotto("ED241", "Scarpe", 60.00, 50));
        Assert.That(_scontrino.getEntryAcquisto("AB123").getNome(), Is.EqualTo("Bagnoschioma"))
        Assert.That(_scontrino.ottieniTotale(), Is.EqualTo(69.20));
        Assert.That(_scontrino.ottieniTotaleScontato(), Is.EqualTo(34.20));
    }
}

```

DEPLOYMENT

ARTEFATTI



DEPLOYMENT TYPE-LEVEL

